SPECIFICATION

Amend paragraph beginning at page 1, line 27, as follows:

As noted above, the backup path protects the primary path against a predetermined set of failures, where each failure is specified by a set of links that may fail simultaneously. However, in such a dynamic environment bandwidth efficiencies can be achieved by providing for the sharing of backup paths. This is possible because, by assumption, the failure scenarios against which restorability is to be maintained [[is]] <u>are</u> known. Therefore, it is possible to determine whether two given links, or nodes, can fail at the same time. If they cannot fail at the same time, backup bandwidth for these elements is shareable. Since bandwidth can be shared along the backup links, the amount of bandwidth reservation to be done on each link in the backup path must be determined.

Amend heading at page 2, line 21, as follows:

BRIEF DESCRIPTION OF THE [[DRAWING]] <u>DRAWINGS</u>

Amend paragraph beginning at page 5, line 9, as follows:

With respect to the backup path, each link $e \in Q$ performs steps 350 through 380, shown in FIG. 3. In step 350, each link on the backup path checks if the primary path and the backup path can fail simultaneously. For this, each link on the backup path accesses the above-mentioned *failure information* (stored at each link). In particular, for each $f \in F$, a link computes $\cancel{L_f \cap P \cap \{e\}}$ $\underline{L_f \cap P \cap \{e\}}$. If $\cancel{L_f \cap P \cap \{e\} \neq \emptyset}$ $\underline{L_f \cap P \cap \{e\} \neq \emptyset}$ for some $f \in F$, then the demand is rejected (in step 355) since the primary path and the backup path can fail simultaneously. On the other hand, if the primary path and the backup path cannot fail simultaneously, then each link on the backup path checks if the primary path can fail in step 360. In particular, for each $f \in F$, a link computes $L_f \cap P$. If the primary path does not fail, no updating is necessary, and the demand may be accepted in step 385. However, if the primary

path can fail, i.e., if $L_f \cap P \neq \emptyset$, then the respective usage information for that link is updated in step 365. In particular, each link in the backup path computes $D_f^e = D_f^e + d$, and computes the new backup reservation, B_e, i.e., $B_e = \max_{f \in F} D_f^e$. In step 370, each link on the backup path checks to see if the new backup reservation amount, B_e, can be reserved (i.e., does link e have the bandwidth available). If the new backup reservation amount, B_e, cannot be reserved, then the demand is rejected in step 380 (and the values of the usage information - changed in step 365 - are returned to their previous values). Similar to rejections of the demand on the primary path, any rejections on the backup path require the source node to re-compute alternative routes. Otherwise, the demand is accepted in step 375 and the backup path can be shared in accordance with the principles of the invention.